



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Geocoding of trees from street addresses and street-level images

Citation for published version:

Laumer, D, Lang, N, van Doorn, N, Mac Aodha, O, Perona, P & Dirk Wegner, J 2020, 'Geocoding of trees from street addresses and street-level images', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 162, pp. 125-136. <https://doi.org/10.1016/j.isprsjprs.2020.02.001>

Digital Object Identifier (DOI):

[10.1016/j.isprsjprs.2020.02.001](https://doi.org/10.1016/j.isprsjprs.2020.02.001)

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Peer reviewed version

Published In:

ISPRS Journal of Photogrammetry and Remote Sensing

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Geocoding of trees from street addresses and street-level images

Daniel Laumer^a, Nico Lang^a, Natalie van Doorn^b, Oisín Mac Aodha^c, Pietro Perona^d,
Jan Dirk Wegner^a

^a*EcoVision Lab, Photogrammetry and Remote Sensing, ETH Zürich, Switzerland*

^b*Forest Service, US Department of Agriculture, USA*

^c*University of Edinburgh, UK*

^d*California Institute of Technology, USA*

Abstract

We introduce an approach for updating older tree inventories with geographic coordinates using street-level panorama images and a global optimization framework for tree instance matching. Geolocations of trees in inventories until the early 2000s were recorded using street addresses whereas newer inventories use GPS. Our method retrofits older inventories with geographic coordinates to allow connecting them with newer inventories to facilitate long-term studies on tree mortality etc. What makes this problem challenging is the different number of trees per street address, the heterogeneous appearance of different tree instances in the images, ambiguous tree positions if viewed from multiple images and occlusions. To solve this assignment problem, we (i) detect trees in Google street-view panoramas using deep learning, (ii) combine multi-view detections per tree into a single representation, (iii) and match detected trees with given trees per street address with a global optimization approach. Experiments for > 50000 trees in 5 cities in California, USA, show that we are able to assign geographic coordinates to 38 % of the street trees, which is a good starting point for long-term studies on the ecosystem services value of street trees at large scale.

Keywords: Geocoding, global optimization, deep learning, image interpretation, object detection, Faster R-CNN, urban areas, street trees, tree inventories, city scale, Google Street View

1. Introduction

An urban forest inventory represents a snapshot in time of trees in a given geographic area (Roman et al., 2013). Urban tree inventories are gaining in popularity and increasingly entering the digital age, with digital devices replacing index cards, and data posted on websites facilitating public access and management (Twardus and Nowak, 2018). Of particular interest to municipalities are street tree inventories since street trees lie within their management scope and are a publicly visible component of the urban forest. Street tree inventories may be used for directing management priorities and assessing pest and pathogen risk (Bond, 2013), surveying species diversity and size distribution (McPherson and Kotow, 2013), and estimating ecosystem services (McPherson et al., 2005, 2016). When updated over time, inventories can become key components of long-term monitoring data,

which are essential for understanding changes in urban forests (Roman et al., 2013). Repeated inventories have been used by researchers to explore trends in street tree populations such as changes in species composition (Dawson and Khawaja, 1985), tree health impacts of storms (Hallet et al., 2018), estimates of demographic rates (Roman et al., 2013, 2014; Widney et al., 2016), as well as determinants of mortality or growth (Koeser et al., 2013; van Doorn and McPherson, 2018). Analyses of repeated tree measurements may help predicting population stability and identify trends. Baseline estimates of demographic rates such as mortality or replacement rates provide a basis for comparison to future measurements. Having information on how the urban forest is changing allows managers to identify the more vulnerable segments of the tree population and to focus management efforts. For example, areas of higher mortality can imply a need for maintenance, increased replacement planting, or a change in the species palette. Spatially-explicit tree data are important for several reasons (Alonzo et al., 2016): ecosystem functions may vary throughout a city due to legacy effects (Roman et al., 2017), species may have different threats from pests, diseases, and fire (Santamour, 1990; Laćan and McBride, 2008), and distribution of ecosystem services may be contributing to environmental injustices (Landry and Chakraborty, 2009). Geographic coordinates are key pieces of information for long-term monitoring studies because they allow for the same tree to be tracked and measured through time with some degree of certainty (van Doorn et al., in press).

For modern inventory collection, trees are typically mapped with GPS devices and thus come with a fairly accurate geographic coordinate, making repeated measurements much easier. However, before the prevalence and ease of GPS, trees in inventories were typically only referenced by street address. Manually linking such tree inventories to GPS-measured inventories is infeasible at a large scale. As a result, a lot of valuable inventory information remains unexplored because there is no automated way, yet, to link those two types of inventories.

Our main motivation is to provide a tool for building long-term tree data sets that can be used to assess changes in the urban forest and corresponding changes in ecosystem services. Although street trees make up a small proportion of the urban forest, they provide considerable ecosystem services (McPherson et al., 2016), and disservices to the urban landscape (Escobedo et al., 2011; Pataki et al., 2011). Benefits include improvement in air quality, a reduction of the heat island effect, increased carbon capture and storage, rising property values, and an improvement in individual and community wellbeing (Nowak et al., 2002; McPherson et al., 2016). According to the most recent estimate (McPherson et al., 2016) there are 9.1 million trees lining the streets of California with an ecosystem services value of \$1 billion per year or \$111 per tree, i.e. \$29 per inhabitant of California. However, little information is available on how street tree populations are changing in terms of demographic rates. While static data sets such as canopy cover and plot data can provide a snapshot in time, demographic details such as growth rates and mortality rates, as well as changes in tree health through time require assessment of the same trees more than once. Many municipalities do not have the funding to complete repeat inventories and/or may place little priority on matching individual trees to create a time series. Thus, making accessible the information in legacy inventories is of high interest to both researchers interested in

quantifying tree demographic rates and ecosystem services and for urban forest practitioners interested in data-driven management (e.g., knowing where there are hotspots of tree mortality, or high growth).

The hope is that our approach on retrofitting existing street tree inventories with geographic coordinates will enable large-scale longitudinal studies, where data of the same population of trees can be analysed over decades. Emphasis of our approach is thus on assigning geographic coordinates to a high absolute number of trees across the state of California as opposed to geocoding each individual tree of the given data base correctly.

In order to geo-code hundreds of thousands of street trees across California with corresponding addresses, we analyse publicly available street-level panoramas across the whole state. For this purpose, we propose to start from a simplified, computationally more efficient version of (Wegner et al., 2016; Branson et al., 2018) for tree detection and geo-localization. We replace the complex and computationally costly conditional random field formulation with a weighted average that condenses multiple detections of the same tree from different views into a single one. Tree instances detected in images are matched to inventory entries that come with street addresses. Preparing and training the system basically consists of manually labeling trees in images (four days), training the tree detector (four hours), and restructuring all inventories into a homogeneous format (one day). Our system comes at virtually no cost if we put aside costs for running computers and downloading Google images. Given appropriate hardware and a fast internet connection for downloading images, the method scales to arbitrarily large data sets.

2. Related Work

Municipal tree inventories are traditionally collected by urban forestry professionals (either municipal staff or contracted-out tree management company arborists) but more recently, some programs also have significant citizen science components (Cozad et al., 2005; Bloniarz and Ryan, 1996; Roman et al., 2017). Variables collected in tree inventories vary by program goal, but the common minimum dataset typically includes tree species, condition, and location (Östberg, 2013; van Doorn et al., in press). The most common method to record tree location is by street address (Roman et al., 2013). Street addresses, while a viable alternative when no technology is available (van Doorn et al., in press), has known drawbacks (Crown et al., 2018). For example, NYC Parks found that the street address method lacked precision, and created confusion at large building sites where multiple trees were linked to one address. In 2005, NYC Parks made an effort to switch from street addresses to geocoded locations, but 23000 locations could not be geocoded because of incorrect location data (Crown et al., 2018), suggesting that the link between the old datasets and new ones cannot be made, resulting in a break in continuity of the dataset.

Cities and communities interested in maintaining their tree inventories are tasked with converting approximate tree locations from older inventories to more accurate locations from modern geocoding methods. However, as methods of data collection are refined, there is a risk that older inventories without accurate location data may be unusable for urban tree monitoring. One approach to translating past municipal inventories includes georeferencing

via GPS and on-screen Geographic Information Systems, as exemplified in Ithaca, NY (City of Ithaca, 2018). This approach may be feasible for smaller cities but may be too costly and time consuming for larger cities with more trees.

Tree detection in an automated way, especially in forests, has attracted much attention in the recent 30 years (see Larsen et al. (2011) and Kaartinen et al. (2012) for a detailed comparison of methods until 2012). More recent works usually rely on airborne LiDAR (Lähivaara et al., 2014; Zhang et al., 2014; Kwak et al., 2017; Matasci et al., 2018), a combination of LiDAR and aerial imagery (Qin et al., 2014; Paris and Bruzzone, 2015), only aerial imagery (Yang et al., 2009), high-resolution satellite imagery (Li et al., 2016), or medium resolution satellite imagery where tree identification and counting is cast as a semantic density estimation problem (Rodriguez and Wegner, 2018). Automated detection of trees in cities has received less attention. Straub (2003) designed a hierarchical workflow by first segmenting aerial images and height models into consistent regions at multiple scales and then refining tree crown shapes with active contours. Lafarge and Mallet (2012) produced a seminal work in the urban environment, creating 3D city models from dense aerial LiDAR point clouds. Besides trees, the authors reconstructed buildings and the ground. After an initial semantic segmentation with a breakline-preserving markov random field (MRF), 3D templates consisting of a cylindrical trunk and an ellipsoidal crown were fitted to the data points. Jaakkola et al. (2010) also modeled tree trunks as cylinders with LiDAR point clouds acquired with an unmanned aerial vehicle (UAV) whereas Monnier et al. (2012) applied a similar approach to LiDAR data acquired with a moving vehicle on the ground. For completeness we note that there is a lot more research on vegetation species mapping but we are not dealing with it in this work and thus a full review is beyond the scope of this paper. We kindly refer the reader to (Alonzo et al., 2014; Wegner et al., 2016; Liu et al., 2017; Branson et al., 2018; Hartling et al.; Aval et al.; Sidike et al., 2019) for some of the most recent works.

Urban object detection from ground-level imagery is a core research topic in computer vision often approached from an autonomous driving perspective with various existing public benchmarks like KITTI Geiger et al. (2013), CityScapes Cordts et al. (2016), or Mapillary Neuhold et al. (2017). In these scenarios, dense image sequences are acquired with minor viewpoint changes in driving direction with forward facing cameras enabling object detection and re-identification across consecutive views Chen et al. (2017); Zhao et al. (2018). In contrast, we work with existing Google street-view panorama images that come with wide baselines (usually around 50 meters) between image acquisition locations, image stitching artefacts, only coarse, inaccurate pose information, and strong scale change between views. In recent work (Nassar et al., 2019a), we propose a siamese CNN approach that combines visual cues and coarse pose estimation of pairs of images to re-identify objects in street-level images. This work was extended (Nassar et al., 2019b) to a full multi-view object detection approach that learns urban object detection end-to-end using visual appearance cues in images and coarse pose evidence.

Other methods geolocalize ground level images using different techniques to match features to corresponding ground-level images database (Lin et al., 2013; Müller-Budack et al., 2018). For example, ground-level imagery can be combined with aerial imagery to accom-

plish a pixel-accurate semantic segmentation of the scene, or road respectively (Zhai et al., 2017; Mátyus et al., 2016). Lefèvre et al. (2017) detects changes by with siamese CNNs through comparing aerial imagery and street view panoramas warped to aerial image geometry. Cao et al. (2018) classify land use categories in urban areas with semantic features from both aerial and ground-level images. Krylov et al. (2018) geo-localize traffic lights and telegraph poles in street-level images with monocular depth estimation using CNNs. A Markov Random Field triangulates object positions. The same authors extend their approach by adding LiDAR data for object segmentation, triangulation, and monocular depth estimation for traffic lights Krylov and Dahyot (2018). Zhang et al. (2018) propose a CNN-based object detector for poles and apply a line-of-bearing method to estimate the geographic object position. In this paper, we advocate for a simplified version of (Wegner et al., 2016; Branson et al., 2018) for tree detection and geo-localization that is less costly to compute than a full end-to-end approach (Nassar et al., 2019b) at very large scale (i.e., 48 cities in California).

3. Methods

In this section, we describe our method to assign geographic coordinates to street trees given street addresses and ground-level panorama images. Given a data base of trees with street addresses, our new system presented here does the following (see flowchart in Fig. 1):

- It retrieves approximate geographic coordinates for each address,
- downloads the available street-view panorama images for each address,
- automatically detects all trees per image,
- integrates individual tree detections per image into a single geographic coordinate per tree,
- and finally matches each detected tree with an entry in the data base.

Because tree assignments to street addresses can be ambiguous due to multiple trees per address as well as trees that can be assigned to more than one address, we solve tree matching via an optimization framework. Geo-coded trees detected from images are matched to tree entries in the data base with geo-coded street addresses. By globally minimizing the sum over the misalignment between geographic coordinates of detected trees and geographic coordinates of street addresses of trees in the inventory, we seek a globally optimal solution per city. In order to avoid having to match all possible pairs of detected trees and tree entries per inventory, we introduce simplifying assumptions that take into account the locally constrained nature of the matching problem.

3.1. Geocoding of street addresses

We reformat all street addresses collected manually in situ into a unified representation that can be processed by the Google geocoding API. Approximate coordinates of street

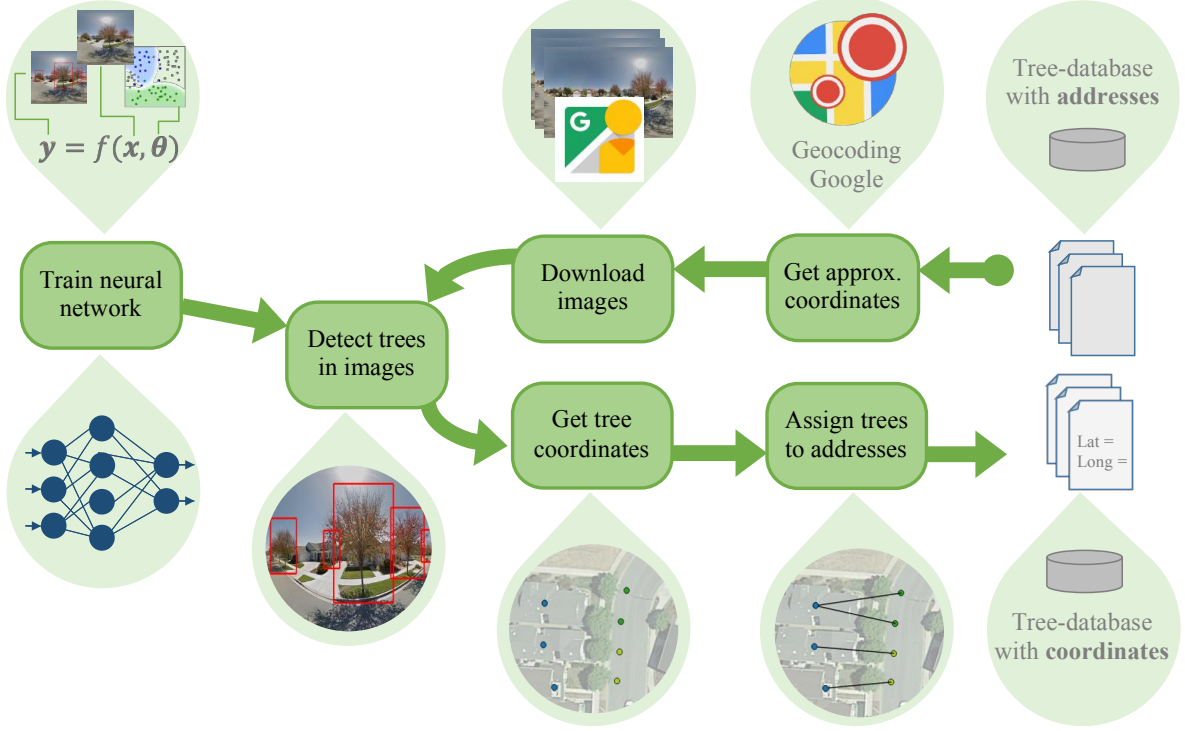


Figure 1: Overview of the workflow for assigning geographic coordinates to trees of an inventory data base given street addresses and street-level panorama images.

addresses are generated using a Google API¹, which returns geo-coordinates centered on the building of a particular parcel (i.e., portion of land attributed to a certain street address). In order to detect outliers before further processing (i.e., addresses projected far away from the rest of the coordinates), we rely on the z-score for latitude $z_{i,lat}$ and longitude $z_{i,long}$. The z-score compares the position of each individual latitude lat_i and longitude $long_i$ to the average values μ_{lat} and μ_{long} across all coordinates $i = 1, 2, 3, \dots, n$ (with n being the total number of street addresses in the whole set) in terms of standard deviations σ_{lat} and σ_{long} (Eq. 1). Note that in our case, a set is equal to a tree inventory of an individual municipality and outliers are thus tree positions projected beyond the boundaries of a municipality.

$$z_{i,lat} = \frac{lat_i - \mu_{lat}}{\sigma_{lat}}, \quad z_{i,long} = \frac{long_i - \mu_{long}}{\sigma_{long}} \quad (1)$$

¹<https://developers.google.com/maps/documentation/geocoding/intro>

with

$$\mu_{lat} = \frac{1}{n} \sum_{i=1}^n lat_i, \quad \sigma_{lat} = \sqrt{\frac{1}{n} \sum_{i=1}^n (lat_i - \mu_{lat})^2} \quad (2)$$

outlier if $z_{i,lat} > 3$ OR $z_{i,long} > 3$

All points with a z-score above a certain threshold are discarded. We empirically found that coordinates further away from the set than three times the standard deviation are outliers and thus set the threshold to 3.

It should be noted that the Google geocoding API uses different ways and accuracy levels to geocode street addresses. In 68% of our cases, it returns geo-coordinates that are located in the center of the biggest building on the parcel at a particular street address. For all remaining cases where rooftop coordinates are unavailable, less accurate approximations are made. Approximations are usually computed by interpolating between junctions or along a polyline or region. Resulting coordinates are often located on the road in front of the parcel instead of being positioned at the center of the biggest building. It turns out that these cases complicate the matching process, because the geocoded coordinates of street addresses on either road side are located in very close proximity leading to a higher risk of tree and address mismatch. We investigate how this affects our results in the experimental section.

3.2. Tree detection

For tree detection, we download Google street-view panoramas within a 50 meter radius around each address coordinate, detect all trees in the panoramas, project each individual detection to geo-coordinates and combine all individual, geocoded detections per tree into a single geographic location. Trees in street-view panorama images are detected with the Faster R-CNN method (Ren et al., 2015), an object detector using deep convolutional neural networks which we already applied successfully in our recent works (Wegner et al., 2016; Branson et al., 2018). The output of the tree detector is one bounding box per detected tree in each street-view panorama (see example in Fig. 2). We project all bounding boxes

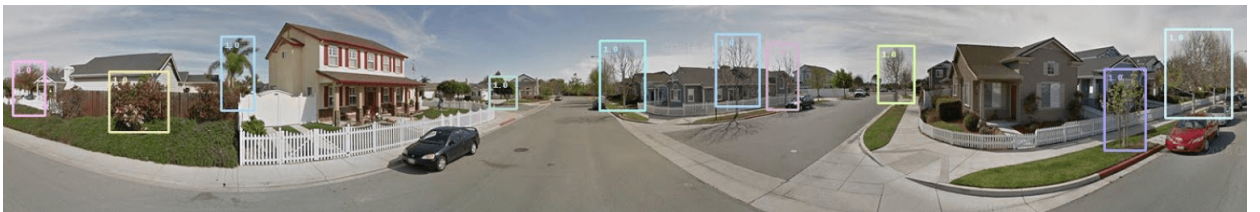


Figure 2: Bounding boxes around automatically detected trees in the center slice of an example street-view panorama. (Imagery © 2019 Google)

of detected trees per panorama to geographic coordinates assuming locally flat terrain with the equations given in Wegner et al. (2016) and Branson et al. (2018). More precisely, we assume that the terrain is locally flat and that the camera is levelled against the ground (pitch = 0). This assumption implies that the horizon is always in the horizontal center-line

of the panorama. We empirically found that this simplifying assumption holds for most cases except roads along high slopes, where trees on the road side are positioned either above or below the road surface. Another exception is road surfaces with high degrees of tilt. However, we did not face any of these complex scenarios in our data set because it does not include municipalities in steep mountainous areas.

In order to compute the position of a tree in geo-coordinates from a single ground-level panorama image, we need to estimate its distance from the given camera position and its direction (camera position and heading come with the panorama meta-file). We take the vertical center axis of a detected bounding box as reference for the direction. With an average camera height above ground of 3 meters, we project each individual detection to geo-coordinates with the center point of the bottom line per bounding box. Projection of tree bounding boxes to geo-coordinates is performed for each detected tree per image. With an average spacing of 15 meters between panoramas, each tree is usually detected multiple times.

We propose a faster to compute, less complex, and easily scalable method of our original tree detection method (Wegner et al., 2016; Branson et al., 2018). The conditional random field of (Wegner et al., 2016; Branson et al., 2018) is replaced with a simple weighted average. More precisely, we compute a score per tree detection by inverse distance weighting of the object detection scores. We select the detection with the highest score per neighborhood and sample all tree detections falling within a four meter radius around it. All coordinates of this set of tree detections are combined in a weighted average with the weight being the distance weighted scores.

A positive side-effect of this procedure is that trees located far away from the street, usually located on private land and not part of the public street-tree population, are filtered out, too. Example results of this approach are shown overlaid on an aerial image in Fig. 3 and projected back into the four individual street-view panoramas (color encodes camera) in Fig. 4.

3.3. Matching detected trees to inventory entries

We cast the task of matching detected trees to tree entries in the database as an optimization problem where the total distance between all pairs of trees is minimized across an entire municipality. In order to cope with missing detections caused by, for example, occlusions in the image, missing street-view panoramas or simply missed trees by the detector, we set a maximum threshold of $M = 50 \text{ meters}$. In case the distance between a tree detected in the street-view panoramas and the geographic coordinate of the street address from a tree of the database exceeds this threshold, no tree is assigned. Our algorithm works as follows:

- Given n number of geocoded street addresses and m number of geocoded detected trees, we compute the distance $dist_{i,j}$ between each address-tree pair ($i \in \{1, 2, \dots, n\}$, $j \in \{1, 2, \dots, m\}$).
- We define $x_{i,j}$ as a binary decision variable that indicates if an address i is assigned to tree j .

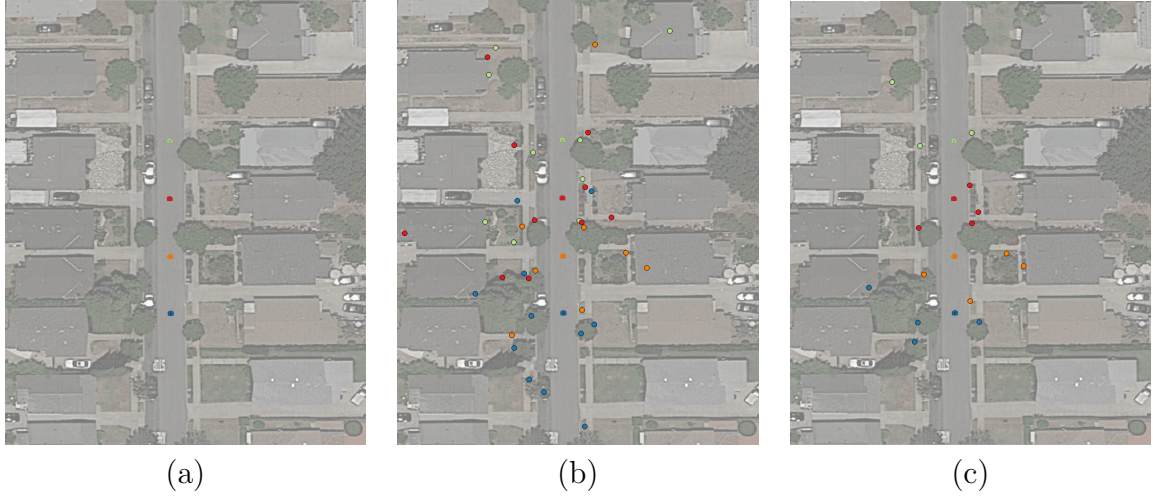


Figure 3: Geocoding of tree detections with inverse distance weighting of detection scores combining multiple detections per individual tree into a single position in geographic coordinates. (a) positions of street-view cameras, (b) all tree detections projected to geographic positions (color encodes camera), (c) final tree positions after non-maximum suppression. (Imagery © 2019 Google)

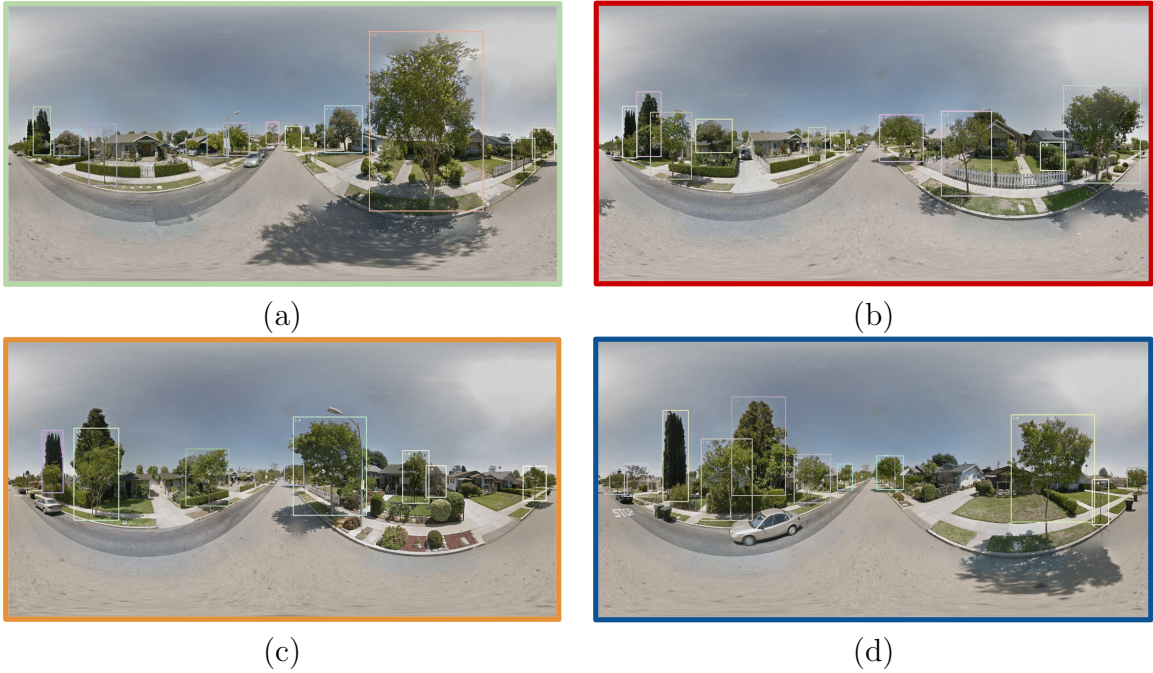


Figure 4: Tree detection results of the example shown in Fig. 3 projected back into the four individual street-view panoramas (color encodes camera). (Imagery © 2019 Google)

- Our objective function to be maximized is: $\max \sum_i \sum_j x_{i,j} \times (M - dist_{i,j})$ with M the distance threshold to avoid matching trees to database entries that are spaced too far apart.
- We introduce constraints to ensure that only one street address is assigned per tree

$\sum_j x_{i,j} \leq 1$ and that the number of assigned detected trees per street address does not exceed K , the total number of trees at that address given in the database $\sum_i x_{i,j} \leq K$.

Although our approach strives for minimizing address-to-tree distances globally for an entire city, most tree assignment cases are locally constrained. This is also reflected by threshold M that discards pairs of distance larger than 50 *meters*. However, distance computation between all possible pairs per data set is already very time consuming, needs a lot of memory, and neglects the local nature of the assignment problem. Therefore, we avoid explicit distance computation and compute a subset of tree candidate matches in the local vicinity of each street address via thresholding geographic latitude and longitude. This procedure leads to a sparse matrix representation of the assignment problem, where distances are only computed between a street address and detected trees inside a local region of interest. We finally minimize the discrepancy between all street addresses and all detected trees globally using Linear Programming.

4. Results and Discussion

We ran experiments with tree inventories from five cities in the state of California, USA, that come with both accurate geocoordinates and street addresses, which allows us to train, validate, and test our approach. The total number of input trees with street addresses is 57938 and the number of street trees per city varied between 599 (Brentwood) and 34585 (Palo Alto). The total number of trees per municipality is reported in the top row of Tab. 1.

All street tree inventories were originally collected by different companies and stakeholders. This created a lot of inconsistent notation across the original data sets. In a first step, we thus preprocessed all tree inventory files with street addresses into the same format. Geo-coordinates of trees were originally recorded in the local Californian reference system CCS83 and needed to be transformed to WGS84, which we used for tree detections.

In order to train the tree object detector (Ren et al., 2015), we manually labeled 6783 individual tree instances in 718 street-view panoramas with a newly developed labeling tool (Fig. 5). We sampled data locations for manual labeling by randomly selecting trees from inventories. All panoramas within a 10 *m* radius around the geographic position of a tree were downloaded for labeling, which resulted in one to four panoramas per selected tree. All trees in all panoramas were then labeled manually as shown in Fig. 5 with our labeling tool².

4.1. Tree geo-coding results

In this section, we present quantitative results for automatically assigning geographic coordinates to an inventoried tree. We also analyze and discuss failure cases in detail. In total, our approach correctly assigned geographic coordinates to 22363 individual street trees out of 57938 input trees with street addresses for five different municipalities where we have geographic ground truth positions. Results for the full processing pipeline are shown in Fig. 6 overlaid on aerial images of three sample sites. Quantitative results including reasons for errors are reported in Tab. 1 and visually shown as bar chart plots in Fig. 7.

²The labeling tool will be made publicly available upon publication of this paper.

	Arroyo Grande	Brentwood	Burlingame	Palo Alto	Walnut Creek	TOTAL
Tree number	1701	599	13617	34585	7436	57938
Geocoding not possible	4 (0.2)	5 (0.8)	3 (0.0)	1209 (3.5)	0 (0.0)	1221 (2.1)
Geocoding outlier	0 (0.0)	2 (0.3)	180 (1.3)	165 (0.5)	3 (0.0)	350 (0.6)
Geocoding wrong	343 (20.2)	160 (26.7)	3600 (26.4)	6722 (19.4)	2474 (33.3)	13299 (23.0)
No tree detected	170 (10.0)	71 (11.9)	987 (7.2)	3364 (9.7)	587 (7.9)	5179 (8.9)
No tree assigned	65 (3.8)	2 (0.3)	131 (1.0)	159 (0.5)	128 (1.7)	485 (0.8)
Tree assigned incorrectly	393 (23.1)	232 (38.7)	2886 (21.2)	9336 (27.0)	2194 (29.5)	15041 (26.0)
Tree correct	726 (42.7)	127 (21.2)	5830 (42.8)	13630 (39.4)	2050 (27.6)	22363 (38.6)

Table 1: Total number of errors per category and as percentage of the total number of inventory trees in each municipality (in parentheses)

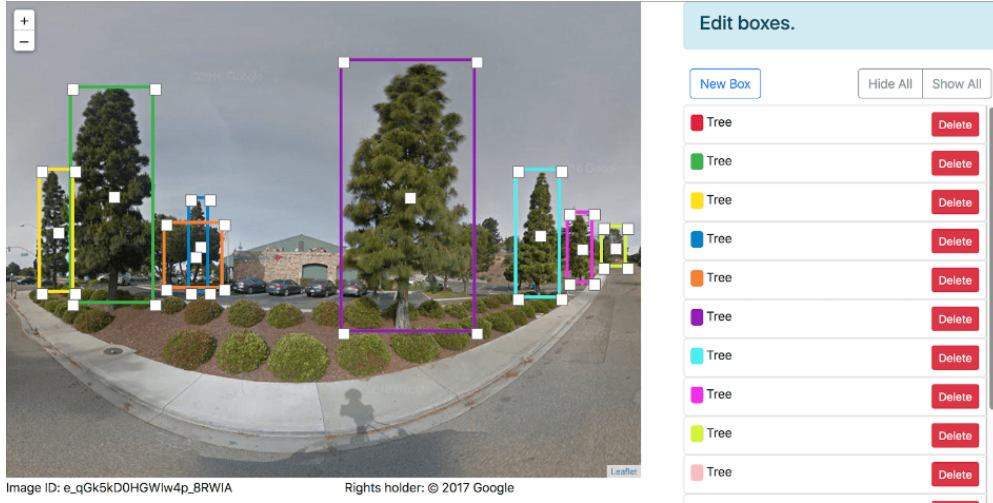


Figure 5: Visualization of our labeling tool for manually drawing bounding boxes around individual trees in street-view panoramas where colors indicate different tree instances. (Imagery © 2019 Google)

One reason for failure is impossible geocoding of a street address in the original inventory database (2.1% of all cases; yellow triangle in Fig. 6, yellow in bar plot Fig. 7). This error occurs if the Google geocoding API cannot match a street address as presented in the original inventory database to a geographic coordinate, which is usually caused by incorrectly spelled or recorded addresses. A more severe failure case is incorrect geocoding of a given street address, i.e. a geographic position far away from the true position is assigned. We distinguish between two geocoding error types: geocoding outliers (0.6% of all cases, black triangle in Fig. 6, black in bar plot Fig. 7) and geocoding farther then 50 meters from the true position (23.0% of all cases, purple star in Fig. 6, red in bar plot Fig. 7). Geocoding outliers are cases where a street address is projected far away from all other entries in the database, as detected via the z-score, described in Sec. 3.1. Reasons include incorrectly spelled or recorded addresses. Geocoding that deviates more than 50 meters from the true position can also be caused by incorrect spelling or recording of a street address. Another reason is the way the Google geocoding API works. Most street addresses ($\approx 68\%$ of all cases) are geocoded via rooftop coordinates of the center of the main building at a given street address. If these are unavailable, the API estimates a geographic coordinate. This is done by either interpolating between two precise points like road intersections ($\approx 27\%$) or, if impossible, by returning the geometric center of a street or a known region ($\approx 5\%$).

Another case is if no tree is detected at a given inventory tree location, either due to missing streetview panorama or a missed detection (8.9% of all cases, blue pentagon in Fig. 6, blue in bar plot Fig. 7). Furthermore, it can happen that all detected trees in the vicinity of an inventory tree have been assigned to neighboring tree entries during the assignment step (0.8% of all cases, purple square in Fig. 6, purple in bar plot Fig. 7).

Finally, when detected trees are matched to database entries, there are erroneous assignments. 26.0% of all inventory database entries received incorrect tree assignments (orange in barplot Fig. 7). To determine if an assignment is correct, we check if a detected tree is

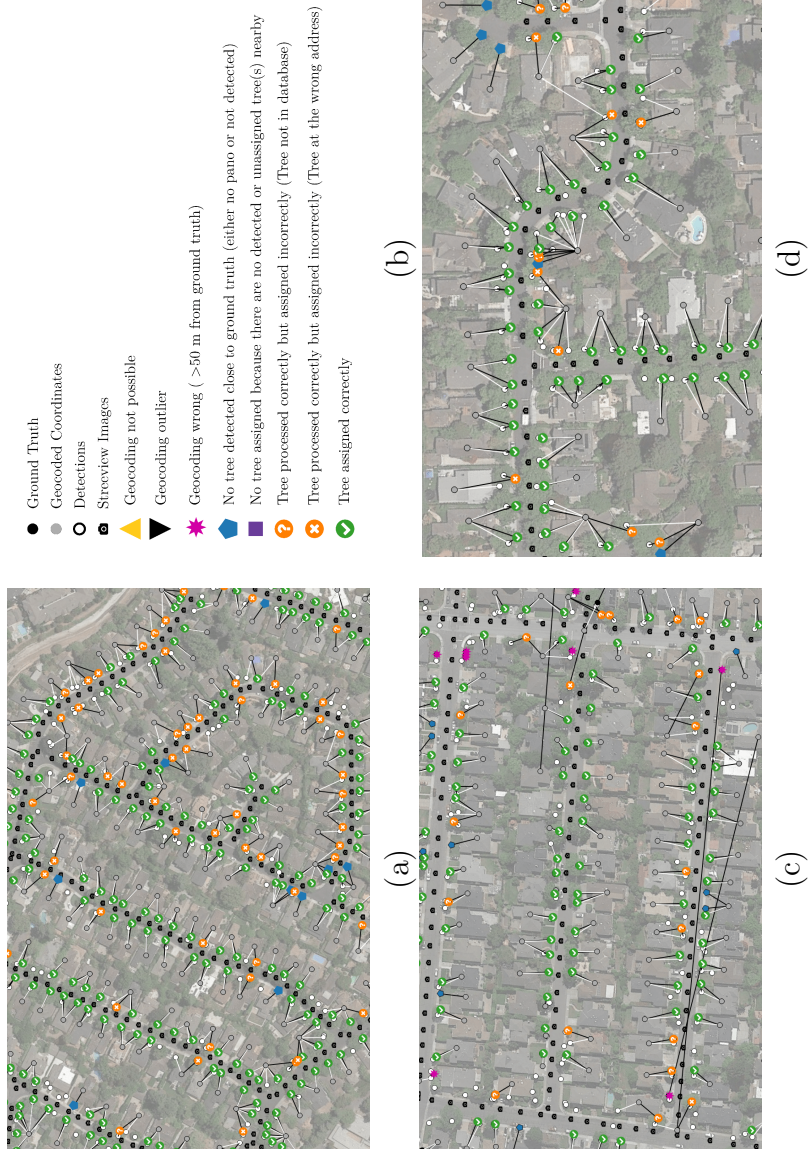


Figure 6: (a,c,d) Results of the full processing pipeline overlaid on aerial images from Google Maps. (b) Legend of symbols used to represent correct cases and different error reasons. We can observe that most errors are trees that are processed correctly but assigned to a wrong address (orange dot with white x) as shown quantitatively in the second-to-last row “Tree assigned incorrectly” of Tab. 1 (Imagery © 2019 Google)

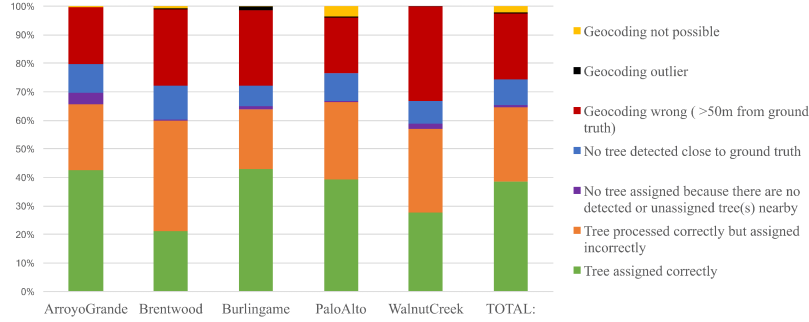


Figure 7: Module errors with respect to total number of inventory trees. The most dominant sources of error are wrong geocoding (red) and correctly processed trees that are assigned to the wrong address (orange) as shown quantitatively in Tab. 1.

matched to the correct geocoded address. Note that we could not check if each individual tree was matched correctly in case of multiple trees per address (83% of all trees share a street address with more trees). Although the database contains information about tree species and height, too, which would in theory allow to distinguish neighboring trees, the vast majority of neighboring trees turned out to be of the same species with the same height.

There are several reasons for assigning a wrong tree to an entry in the database (orange in bar plot Fig. 7). Since our assignment optimizer rewards spatial proximity between detected tree and geocoded street address of the inventory tree entry, a tree close to the boundary between two parcels can be assigned to the wrong one (orange dot with white x in Fig. 6). Errors can occur as a result of detected trees that were originally not recorded in the database (orange dot with white question mark in Fig. 6). This can happen because the trees were originally missed during the field campaign, a new tree was planted, or a tree on private land is detected (i.e., and thus was not recorded in the inventory of public street-trees). If these trees are matched to an entry in the inventory database, they also count as incorrect assignments.

4.2. Error analysis

In this section we analyse in detail the dominant reasons for failure. We look at the influence of geocoding quality, erroneous or missing tree detections, and the assignment step.

Geocoding quality has a big impact on the number of correctly assigned trees. The best possible kind of geocoding are rooftop coordinates in the center of the main building per address as compared to more approximate methods such as the ones described earlier that for example interpolate between road intersections. We plot the proportion of rooftop coordinates per municipality versus the proportion of correctly assigned trees in Fig. 8. In general, a high proportion of successful street address geocoding via rooftop coordinates usually leads to higher numbers of correctly assigned trees. Palo Alto and Arroyo Grande have the highest proportion of rooftop coordinates with $\approx 88\%$ and the lowest proportion of geocoding errors (19.4% and 20.2%). Burlingame looks like an exception at first glance, having a low number of rooftop coordinates, but highest number of correctly assigned trees

(42.8%). However, a look into Tab. 1 reveals that the geocoding error was relatively high (26.4%). In contrast, large parcels with very large buildings like the one shown in Fig. 9

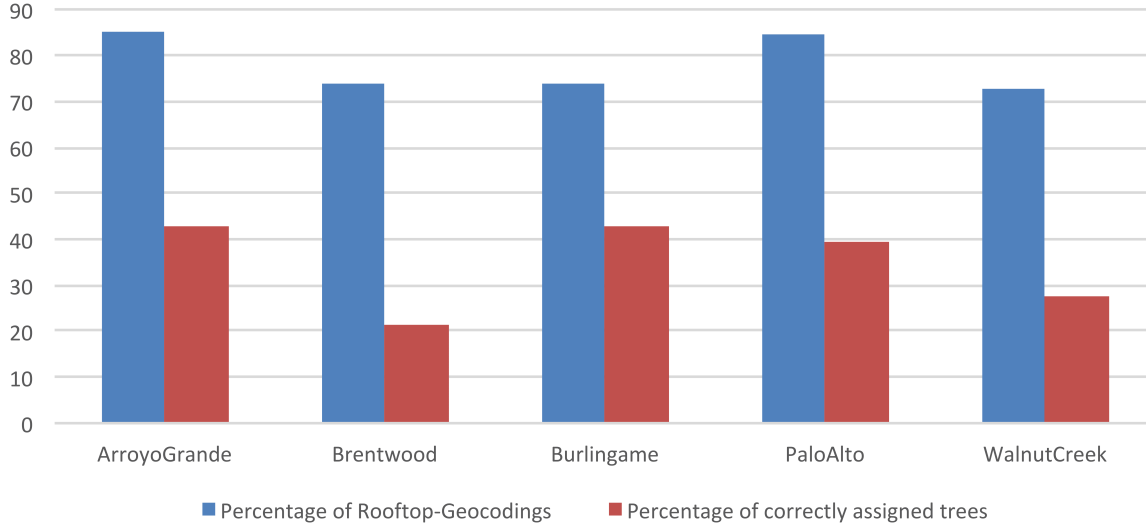


Figure 8: Comparison between geocoding via rooftop coordinates and correctly assigned trees. Higher percentages of rooftop-geocodings (blue) lead to higher percentages of correctly assigned trees (red).

led to a very high amount of errors for several reasons. Large parcels (and buildings) led to large distances between street-trees and rooftop coordinates. Consequently, geocoding is evaluated incorrectly because the rooftop coordinate is farther than 50 meters from the tree position at that street address (purple star in Fig. 9).

Tree detections in Google street-view panoramas can sometimes also fail either due to missing panoramas or an error by the detector. Tree detection failure caused 8.9% of all entries in the inventory database to remain without a valid geo-coordinate (see category *no tree detected* in Tab. 1). One major reason is that at various locations no street-view panorama was acquired close enough to the tree position. An example scene is shown in Fig. 10(a), where black camera symbols indicate acquisition locations of panoramas and blue pentagons indicate database entries without any detected tree. In case panoramas are available, the detector may sometimes miss a tree as is shown with the two blue pentagons in Fig. 10(b). The dominant reason for missed trees (i.e., false negatives) are occlusions. If trees are partially or completely occluded in the panorama by, for example, a truck (Fig. 11(a)), there is no way to recover them and assign a coordinate to a database entry at that address. Another example of a false negative is if a tree in the field is removed between the time that it was inventoried and the time that a street-view image is recorded so the API does not detect a tree but it used to be there. Two examples of false positives, i.e. objects falsely detected as trees, are shown in Fig. 11(b,c). A more detailed analysis of detection error cases and proportions is given in Branson et al. (2018). Note these false detections in individual images usually do not survive the local non-maximum suppression step of our workflow because they are rarely detected in more than one panorama.

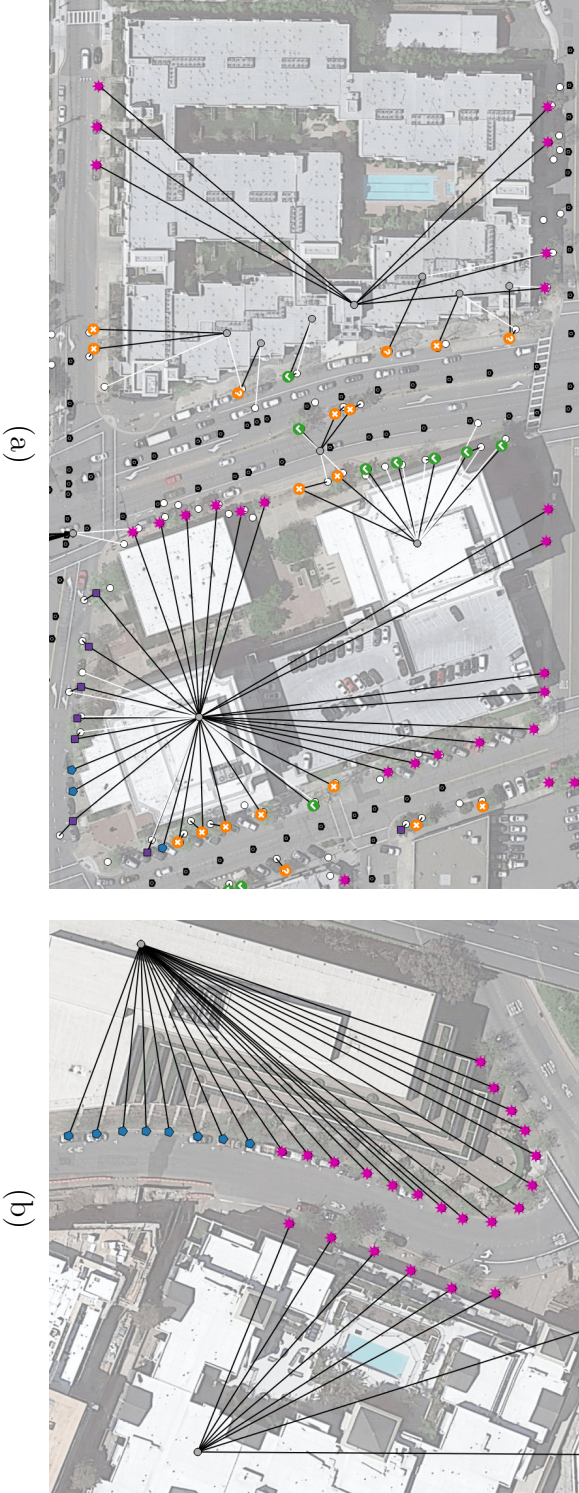


Figure 9: Examples for failure of geocoding (rose stars, refer to legend in Fig. 6(b)): (a) and (b) show street addresses of large buildings with a high number of trees. However, most detected trees are farther away than 50 *meters* (rose star) from the rooftop coordinate and thus count as unassigned. Note that many of these cases are still located close to a tree in the inventory database (white dot) and could be manually processed. (Imagery © 2019 Google)



Figure 10: Examples for typical failure cases of tree detection in panorama images (refer to legend in Fig. 6(b)): (a) no tree detected in the street-view panorama (blue pentagons), (b) no street-view panorama available at junction leading to undetected trees (blue pentagons). (Imagery © 2019 Google)

Trees assigned incorrectly are another major reason for mismatches. Incorrect assignments of detected trees accounted for 26.0% of all entries in the database that were matched incorrectly. Quantitative results in Tab. 1 show that the error distribution and also the total number of successfully assigned trees varied across different municipalities. For example, Brentwood with the lowest number of public street-tree entries in the inventory compared to the four other municipalities, had also the lowest number of correctly assigned trees (21.2.%). The error distribution reveals that a very high percentage of trees is incorrectly assigned. A closer look at Brentwood shows that it is a suburban municipality of relatively low density with many trees close to the street on private land and only very few, scarcely distributed public street-trees. This leads to a high number of cases where trees on private land are wrongly matched to a geocoded inventory entry, often because the tree on private ground is closer to the geographic rooftop coordinate of the main building on a parcel. On the contrary, Burlingame with the highest percentage of assigned trees (42.8%) is a municipality with denser suburban structure and many street-trees on public land that densely line most streets. Incorrectly assigned trees make up only 21.2% of all of Burlingame’s inventory trees as compared to Brentwood’s 38.7%.

4.3. Test on 48 municipalities and discussion

We ran our processing pipeline on all 48 municipalities containing 1100952 input trees across all inventories. The five municipalities that we used for a detailed error analysis because they have geographic coordinates in addition to street addresses, are a subset of these 48 municipalities. We show all results with errors at test time indicated in color in Fig. 12. Our method assigned geographic coordinates to 619229 individual street trees (grey in Fig. 12), which amounted to 56% of all inventory trees. Note that error evaluation at test time (i.e., geographic ground truth position per tree available) can only detect the following failure cases (as shown in Fig. 12):

- geocoding of the street address is impossible (yellow),
- or generates an outlier according to our z-score (black),
- and missing tree detections in the panorama images close to the geocoded street address of a tree in the inventory (purple).

Since we do not have ground truth tree coordinates in this realistic application scenario, we cannot detect all possible failure cases. More precisely, we cannot recognize cases where geocoding is wrong, i.e. geocoding of a given street address passes the z-score but is more than 50 meters away from the true tree position, and where trees are assigned incorrectly. Major reasons for incorrect geocoding are approximate geocoding (no rooftop coordinates, but interpolation between road intersections, for example) and very large parcels and buildings with a very high number of street-trees (Fig. 9). A simple way to significantly reduce the amount of errors caused by erroneous geocoding is filtering out all trees matched via non-rooftop geocoding. We evaluated this idea on those five municipalities where we have ground truth coordinates (and assume a similar behaviour for all other municipalities without

ground truth coordinates). Elimination of all cases without rooftop coordinates decreases the total percentage of incorrect geocoding across the five municipalities by 11.4 percentage points to 11.6%. The overall correctly assigned tree percentage increases by 8.4 percentage points to 47% (see Tab. 1).

Incorrect assignments of trees due to mixing up trees at the boundaries between two parcels or confusion between trees on private land and street-trees is generally difficult to fix. One direction for automatic detection of these errors would be to train a tree species classifier as was done in Branson et al. (2018) to verify whether the species of the tree in the database does actually match the tree in the images. While this would probably work well to detect cases of confusion between trees on private lawns and street-trees (often different species), it would rarely solve mix-ups between two neighboring street-trees at parcel boundaries because these often have the same species.

5. Conclusion

We have presented a novel approach to assign geographic coordinates to street-trees given street addresses and street-view panorama images. Enhancing existing tree inventories with geographical locations allows to automatically match inventories from different dates and to track trees across longer time spans. Depending on the information that is being recorded in tree inventories (e.g., presence/absence, trunk diameter, vigor), repeated measures make it possible to calculate tree mortality rates and detect changes in tree health. Linking legacy inventories to newer inventories is a cost- and time-effective way to obtain a much larger sample size than is typically available to municipalities and researchers. Urban tree demographic studies have for the most part based their analyses on a small number of samples limited in geographic area. Our hope is that with the proposed approach, the sample of the cities' tree population can be increased tremendously.

Experimental evaluation of our method on five different municipalities with geographic coordinates show that the method is capable of assigning geographic coordinates correctly to 38.6% of the trees in the inventory databases. Application to all 48 Californian municipalities with inventory data demonstrates that we can assign geographic locations to 619229 trees out of 1100952 trees (56%). After a simple post-processing step that filters out all addresses geocoded via less accurate, non-rooftop coordinates, geocoding errors can be reduced by ≈ 11 percentage points and the overall percentage of correctly assigned trees can be increased by ≈ 8 percentage points.

The major shortcoming of the proposed method is that 26% of trees were assigned incorrectly (Tab. 1), which is an error that can hardly be detected during a realistic scenario without available ground truthed tree coordinates. Our future work will thus look into strategies to significantly reduce this type of error by implementing strategies such as adding tree species information to the matching step.

We hope this work encourages further research to generate longer time series (over several decades) of street-tree inventories with the ultimate goal of better understanding street tree population dynamics and corresponding changes in ecosystem services at a very large scale.

6. Acknowledgements

This project was supported by funding from the Hasler Foundation, the US Department of Agriculture-Forest Service, and the Swiss National Science Foundation scientific exchange grant IZSEZ0_185641.

References

- Alonzo, M., Bookhagen, B., Roberts, D., 2014. Urban tree species mapping using hyperspectral and lidar data fusion. *Remote Sensing of Environment* 148, 70–83.
- Alonzo, M., McFadden, J., Nowak, D., Roberts, D., 2016. Mapping urban forest structure and function using hyperspectral imagery and lidar data. *Urban Forestry & Urban Greening* , 135–147.
- Aval, J., Fabre, S., Zenou, E., Sheeren, D., Fauvel, M., Briottet, X., . object-based fusion for urban tree species classification from hyperspectral, panchromatic and ndsm data .
- Bloniarz, D., Ryan, H., 1996. The use of volunteer initiatives in conducting urban forest resource inventories. *Journal of Arboriculture* , 75–82.
- Bond, J., 2013. Best Management Practices Tree Inventories, in: 2nd ed. International Society of Arboriculture, pp. 1–35.
- Branson, S., Wegner, J., Hall, D., Lang, N., Schindler, K., Perona, P., 2018. From google maps to a fine-grained catalog of street trees. *ISPRS Journal of Photogrammetry and Remote Sensing* 135, 13–30.
- Cao, R., Zhu, J., Tu, W., Li, Q., Cao, J., Liu, B., Zhang, Q., Qiu, G., 2018. Integrating aerial and street view images for urban land use classification. *Remote Sensing* 10, 1553.
- Chen, X., Ma, H., Wan, J., Li, B., Xia, T., 2017. Multi-view 3d object detection network for autonomous driving, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1915.
- City of Ithaca, 2018. Tree Inventory/GIS. <https://www.cityofithaca.org/253/Tree-Inventory-GIS>. Online; accessed 5-November-2018.
- Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223.
- Cozad, S., McPherson, G., Harding, J.A., 2005. STRATUM Case Study Evaluation in Minneapolis, Minnesota. . Center for Urban Forest Research, USDA Forest Service, Pacific Southwest Research Station, University of California, Dabbs, CA, USA. URL: <https://www.itreetools.org/resources/reports/Minneapolis%20Case%20Study.pdf>.
- Crown, C., Greer, B., Gift, D., Watt, F., 2018. Every Tree Counts: Reflections on NYCs Third Volunteer Street Tree Inventory. *Arboriculture & Urban Forestry* 44, 49–58.
- Dawson, J., Khawaja, M., 1985. Change in street-tree composition of two. Urbana, Illinois neighborhoods after fifty years: 19321982. *Journal of Arboriculture* 11, 344–348.
- Escobedo, F., Kroeger, T., Wagner, J., 2011. Urban forests and pollution mitigation: Analyzing ecosystem services and disservices. *Environmental Pollution* 159, 2078–2087.
- Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* 32, 1231–1237.
- Hallet, R., Johnson, M., Sonti, N., 2018. Assessing the tree health impacts of salt water flooding in coastal cities: A case study in New York City. *Landscape and Urban Planning* 177, 171–177.
- Hartling, S., Sagan, V., Sidike, P., Maimaitijiang, M., Carron, J., . urban tree species classification using a worldview-2/3 and lidar data fusion approach and deep learning .
- Jaakkola, A., Hyypä, J., Kukko, A., Yu, X., Kaartinen, H., Lehtomäki, M., Lin, Y., 2010. A low-cost multi-sensoral mobile mapping system and its feasibility for tree measurements. *ISPRS Journal of Photogrammetry and Remote Sensing* 65, 514–522.
- Kaartinen, H., Hyypä, J., Yu, X., Vastaranta, M., Hyypä, H., Kukko, A., Holopainen, M., Heipke, C., Hirschmugl, M., Morsdorf, F., Nsset, E., Pitkänen, J., Popescu, S., Solberg, S., Wolf, B.M., Wu, J.C.,

2012. An International Comparison of Individual Tree Detection and Extraction Using Airborne Laser Scanning. *Remote Sensing* 4, 950–974.
- Koeser, A., Hauer, R., Norris, K., Krouse, R., 2013. Factors influencing long-term street tree survival in Milwaukee WI, USA. *Urban Forestry & Urban Greening* 12, 562–568.
- Krylov, V., Kenny, E., Dahyot, R., 2018. Automatic discovery and geotagging of objects from street view imagery. *Remote Sensing* 10, 661.
- Krylov, V.A., Dahyot, R., 2018. Object geolocation using mrf based multi-sensor fusion, in: 2018 25th IEEE International Conference on Image Processing (ICIP), IEEE. pp. 2745–2749.
- Kwak, D., Lee, W., Lee, J., Biging, G., Gong, P., 2017. Detection of individual trees and estimation of tree height using lidar data. *Journal of Forest Research* 12, 425–434.
- Laćan, I., McBride, J., 2008. Pest Vulnerability Matrix (PVM): a graphic model for assessing the interaction between tree species diversity and urban forest susceptibility to insects and diseases. *Urban Forestry & Urban Greening* 7, 291–300.
- Lafarge, F., Mallet, C., 2012. Creating Large-Scale City Models from 3D-Point Clouds: a robust approach with hybrid representation. *International Journal of Computer Vision* 99, 69–85.
- Lähivaara, T., Seppänen, A., Kaipio, J.P., Vauhkonen, J., Korhonen, L., Tokola, T., Maltamo, M., 2014. Bayesian Approach to Tree Detection Based on Airborne Laser Scanning Data. *IEEE Transactions on Geoscience and Remote Sensing* 52, 2690–2699.
- Landry, S., Chakraborty, J., 2009. Street trees and equity: evaluating the spatial distribution of an urban amenity. *Environment and Planning A* 41, 2651–2670.
- Larsen, M., Eriksson, M., Descombes, X., Perrin, G., Brandtberg, T., Gougeon, F.A., 2011. Comparison of six individual tree crown detection algorithms evaluated under varying forest conditions. *International Journal of Remote Sensing* 32, 5827–5852.
- Lefèvre, S., Tuia, D., Wegner, J.D., Produit, T., Nassar, A.S., 2017. Toward seamless multiview scene analysis from satellite to street level. *Proceedings of the IEEE* 105, 1884–1899.
- Li, W., Fu, H., Yu, L., Cracknell, A., 2016. Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. *Remote Sensing* 9, 22–34.
- Lin, T.Y., Belongie, S., Hays, J., 2013. Cross-view image geolocalization, in: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Liu, L., Coops, N., Aven, N., Pang, Y., 2017. Mapping urban tree species using integrated airborne hyperspectral and lidar remote sensing data. *Remote Sensing of Environment* 200, 170–182.
- Matasci, G., Coops, N., Williams, D., Page, N., 2018. Mapping tree canopies in urban environments using airborne laser scanning (ALS): a Vancouver case study. *Forest Ecosystems* 5, 1–9.
- Máttyus, G., Wang, S., Fidler, S., Urtasun, R., 2016. Hd maps: Fine-grained road segmentation by parsing ground and aerial images, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3611–3619.
- McPherson, E., Kotow, L., 2013. A municipal forest report card: results for california usa. *Urban Forestry & Urban Greening* 12, 134–143.
- McPherson, E., Simpson, J., Peper, P., Maco, S., Xiao, Q., 2005. Municipal forest benefits and costs in five us cities. *Journal of Forestry* 103, 411–416.
- McPherson, E., van Doorn, N., de Goede, J., 2016. Structure, function and value of street trees in california, usa. *Urban Forestry and Urban Greening* 17, 104–115.
- Monnier, F., Vallet, B., Soheilian, B., 2012. Trees detection from laser point clouds acquired in dense urban areas by a mobile mapping system, in: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 245–250.
- Müller-Budack, E., Pustiu-Iren, K., Ewerth, R., 2018. Geolocation estimation of photos using a hierarchical model and scene classification, in: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 563–579.
- Nassar, A., Lang, N., Lefèvre, S., Wegner, J., 2019a. Learning geometric soft constraints for multi-view instance matching across street-level panoramas, in: *Joint Urban Remote Sensing Event (JURSE)*, pp. 1–4.

- Nassar, A., Lefèvre, S., Wegner, J., 2019b. Simultaneous multi-view instance detection with learned geometric soft-constraints, in: International Conference on Computer Vision, pp. 6559–6568.
- Neuhold, G., Ollmann, T., Bulò, S.R., Kotschieder, P., 2017. The mapillary vistas dataset for semantic understanding of street scenes., in: ICCV, pp. 5000–5009.
- Nowak, D.J., Crane, D.E., Dwyer, J.F., 2002. Compensatory value of urban trees in the united states. *Journal of Arboriculture* 28, 194–199.
- Östberg, J., 2013. Tree inventories in the urban environment: methodological development and new applications. Ph.D. thesis. Swedish University of Agricultural Sciences, Alnarp, Sweden.
- Paris, C., Bruzzone, L., 2015. A three-dimensional model-based approach to the estimation of the tree top height by fusing low-density lidar data and very high resolution optical images. *IEEE Transactions on Geoscience and Remote Sensing* 53, 467–480.
- Pataki, D., Carreiro, M., J.C., Grulke, N., Jennings, V., Pincetl, S., Pouyat, R., Whitlow, T., Zipperer, W., 2011. Coupling biogeochemical cycles in urban environments: ecosystem services, green solutions, and misconceptions. *Frontiers in Ecology and the Environment* 9, 27–36.
- Qin, Y., Ferraz, A., Mallet, C., 2014. Individual tree segmentation over large areas using airborne LiDAR point cloud and very high resolution optical imagery, in: IEEE International Geoscience and Remote Sensing Symposium.
- Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems (NIPS).
- Rodriguez, A., Wegner, J., 2018. Counting the uncountable: deep semantic density estimation from space, in: German Conference on Pattern Recognition.
- Roman, L., Battles, J., McBride, J., 2014. The balance of planting and mortality in a street tree population. *Urban Ecosystems* 17, 387–404.
- Roman, L., McPherson, E., Scharenbroch, B., Bartens, J., 2013. Identifying common practices and challenges for local urban tree monitoring programs across the united states. *Arboriculture & Urban Forestry* 39, 292–299.
- Roman, L.A., Scharenbroch, B.C., Östberg, J.P., Mueller, L.S., Henning, J.G., Koeser, A.K., Sanders, J.R., Betza, D.R., Jordan, R.C., 2017. Data quality in citizen science urban tree inventories. *Urban Forestry and Urban Greening* 22, 124–135.
- Santamour, F., 1990. Trees for urban planting: diversity, uniformity, and common sense, in: Proceedings of the Seventh Conference of the Metropolitan Tree Improvement Alliance (METRIA), pp. 57–65.
- Sidike, P., Sagan, V., Maimaitijiang, M., Maimaitiyiming, M., Shakoar, N., Burken, J., Mockler, T., Fritschi, F., 2019. dPEN: deep Progressively Expanded Network for mapping of heterogeneous agricultural landscape using WorldView-3 imagery. *Remote Sensing of Environment* 221, 756–772.
- Straub, B.M., 2003. Automatic extraction of trees from aerial images and surface models, in: ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.
- Twardus, A.C.D., Nowak, D., 2018. Urban Forest Health Monitoring: Large-Scale Assessments in the United States. *Arboriculture & Urban Forestry* 34, 341–346.
- van Doorn, N., McPherson, E., 2018. Demographic trends in Claremont Californias street tree population. *Urban Forestry & Urban Greening* 29, 200–211.
- van Doorn, N., Roman, L., McPherson, E., Scharenbroch, B., Henning, J., Östberg, J., Mueller, L., Koeser, A., Mills, J., Hallett, R., Sanders, J., Battles, J., Boyer, D., Fristensky, J., Mincey, S., Peper, P., Vogt, J., in press. Urban Tree Monitoring Standards: Resource Guide. Technical Report. US Department of Agriculture - Forest Service.
- Wegner, J.D., Branson, S., Hall, D., Schindler, K., Perona, P., 2016. Cataloging Public Objects Using Aerial and Street-Level Images - Urban Trees, in: IEEE Conference on Computer Vision and Pattern Recognition.
- Widney, S., Fischer, B., Vogt, J., 2016. Tree mortality undercuts ability of tree-planting programs to provide benefits: results of a three-city study. *Forests* 7, 1–65.
- Yang, L., Wu, X., Praun, E., Ma, X., 2009. Tree detection from aerial imagery, in: ACM GIS’09.
- Zhai, M., Bessinger, Z., Workman, S., Jacobs, N., 2017. Predicting ground-level scene layout from aerial

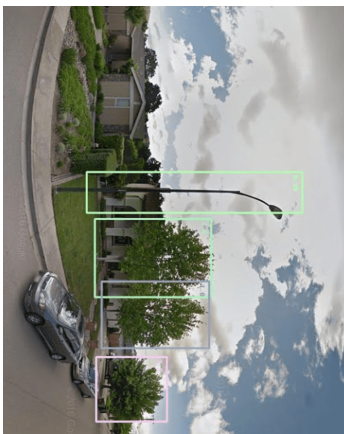
- imagery, in: IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, J., Sohn, G., Brédif, M., 2014. A hybrid framework for single tree detection from airborne laser scanning data: A case study in temperate mature coniferous forests in Ontario, Canada. *ISPRS Journal of Photogrammetry and Remote Sensing* 98, 44–57.
- Zhang, W., Witharana, C., Li, W., Zhang, C., Li, X., Parent, J., 2018. Using deep learning to identify utility poles with crossarms and estimate their locations from google street view images. *Sensors* 18, 2484.
- Zhao, J., Zhang, X.N., Gao, H., Yin, J., Zhou, M., Tan, C., 2018. Object detection based on hierarchical multi-view proposal network for autonomous driving, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–6.



(a)



(b)



(c)

Figure 11: False positives and false negatives of tree detection: (a) tree occluded by a truck, (b) power-pole wrongly classified as tree, (c) street lamp wrongly classified as tree. (Imagery © 2019 Google)



Figure 12: Results for all 48 cities containing in total 1100952 input trees from inventories. Error proportions in each municipality are color-coded by error type. Missing (detected) trees at a geocoded street address are the most dominant reason for errors (purple).